

**IN THE CLAIMS:**

---

1. (Currently Amended) A method for propagating type information for hardware device nodes in a graphical program, wherein the method operates in a computer including a display screen and a user input device, the method comprising:

displaying on the display screen of the computer a first hardware device node in the graphical program in response to user input, wherein the graphical program comprises a plurality of interconnected nodes or icons, wherein the plurality of interconnected nodes or icons visually indicate functionality of the graphical program;

associating the first hardware device node with a hardware device;

displaying on the display screen a second hardware device node in the graphical program in response to user input;

connecting the first hardware device node to the second hardware device node in response to user input;

propagating information from the first hardware device node to the second hardware device node, wherein the information specifies the hardware device with which the first hardware device node is associated, wherein said propagating occurs in response to said connecting the first hardware device node to the second hardware device node;

wherein the graphical program is executable by the computer.

2. (Original) The method of claim 1,

wherein said displaying the first and second hardware device nodes in the graphical program comprises including the first and second hardware device nodes in a block diagram of the graphical program, wherein the block diagram visually indicates functionality of the graphical program.

3. (Original) The method of claim 1, further comprising:

associating the second hardware device node with the hardware device with which the first hardware device node is associated, in response to said propagating the information to the second hardware device node.

4. (Original) The method of claim 1,

wherein said connecting the first hardware device node to the second hardware device node comprises connecting a wire from an output terminal of the first hardware device node to an input terminal of the second hardware device node.

5. (Original) The method of claim 1,

wherein said associating the first hardware device node with a hardware device comprises associating the first hardware device node with a hardware device class corresponding to the hardware device;

wherein said propagating information from the first hardware device node to the second hardware device node comprises propagating information specifying the hardware device class with which the first hardware device node is associated.

6. (Original) The method of claim 5, further comprising:

associating the second hardware device node with the hardware device class, in response to said propagating the information to the second hardware device node.

7. (Original) The method of claim 6, further comprising:

associating the second hardware device node with a method of the hardware device class in response to user input;

wherein during execution of the graphical program the second hardware device node is operable to invoke the method.

8. (Original) The method of claim 6, further comprising:

associating the second hardware device node with a property of the hardware device class in response to user input;

wherein during execution of the graphical program the second hardware device node is operable to perform one or more of: 1) getting the property; and 2) setting the property.

9. (Original) The method of claim 1, further comprising:

executing the graphical program, wherein during execution of the graphical program the second hardware device node is operable to access the hardware device.

10. (Currently Amended) A method for performing type checking for a hardware device node in a graphical program, wherein the method operates in a computer including a display screen, the method comprising:

displaying on the display screen of the computer a first hardware device node in the graphical program in response to user input, wherein the graphical program comprises a plurality of interconnected nodes or icons, wherein the plurality of interconnected nodes or icons visually indicate functionality of the graphical program;

CV associating the first hardware device node with a first hardware device class in response to user input;

selecting a method or property of the first hardware device class for the first hardware device node in response to user input;

changing the first hardware device node to have an association with a second hardware device class in response to user input; and

performing type checking to determine whether the method or property is valid for the second hardware device class, in response to said changing the first hardware device node to have an association with the second hardware device class;

wherein the graphical program is executable by the computer.

11. (Original) The method of claim 10, further comprising:

indicating an invalid condition if the method or property is not valid for the second hardware device class.

12. (Original) The method of claim 11,

wherein said indicating the invalid condition comprises altering the visual appearance of a wire connected to an input terminal of the first hardware device node, wherein the wire provides information specifying the second hardware device class with which the first hardware device node is associated.

13. (Original) The method of claim 10, further comprising:  
preventing execution of the graphical program if the method or property is not  
valid for the second hardware device class.

14. (Original) The method of claim 10,  
wherein the first hardware device node has an input terminal for receiving  
information specifying a hardware device class with which to associate the first hardware  
device node;

wherein said associating the first hardware device node with the first hardware  
device class comprises connecting a first wire to the input terminal;

wherein said changing the first hardware device node to have an association with  
a second hardware device class comprises connecting a second wire to the input terminal.

15. (Original) The method of claim 10,  
wherein the first hardware device node is a register access node.

16. (Original) The method of claim 10,  
wherein said performing type checking to determine whether the method or  
property is valid for the second hardware device class comprises:

determining a list of valid methods and properties for the second hardware  
device class; and

determining whether the method or property is included in the list of valid  
method and properties.

17. (Original) The method of claim 16,  
wherein said determining the list of valid methods and properties for the second  
hardware device class comprises determining the valid methods and properties from a  
type library, wherein the type library includes information regarding the second hardware  
device class.

18. (Currently Amended) A memory medium comprising program instructions executable to:

display a first hardware device node in a graphical program in response to user input, wherein the graphical program comprises a plurality of interconnected nodes or icons, wherein the plurality of interconnected nodes or icons visually indicate functionality of the graphical program;

associate the first hardware device node with a hardware device;

display on [[the]] a display screen a second hardware device node in the graphical program in response to user input;

connect the first hardware device node to the second hardware device node in response to user input;

propagate information from the first hardware device node to the second hardware device node, wherein the information specifies the hardware device with which the first hardware device node is associated, wherein said propagating occurs in response to said connecting the first hardware device node to the second hardware device node;

wherein the graphical program is executable by a computer system.

19. (Original) The memory medium of claim 18,

wherein said displaying the first and second hardware device nodes in the graphical program comprises including the first and second hardware device nodes in a block diagram of the graphical program, wherein the block diagram visually indicates functionality of the graphical program.

20. (Original) The memory medium of claim 18, further comprising program instructions executable to:

associate the second hardware device node with the hardware device with which the first hardware device node is associated, in response to said propagating the information to the second hardware device node.

21. (Original) The memory medium of claim 18,

wherein said connecting the first hardware device node to the second hardware device node comprises connecting a wire from an output terminal of the first hardware device node to an input terminal of the second hardware device node.

22. (Original) The memory medium of claim 18,

wherein said associating the first hardware device node with a hardware device comprises associating the first hardware device node with a hardware device class corresponding to the hardware device;

wherein said propagating information from the first hardware device node to the second hardware device node comprises propagating information specifying the hardware device class with which the first hardware device node is associated.

C2  
23. (Original) The memory medium of claim 22, further comprising program instructions executable to:

associate the second hardware device node with the hardware device class, in response to said propagating the information to the second hardware device node.

24. (Original) The memory medium of claim 23, further comprising program instructions executable to:

associate the second hardware device node with a method of the hardware device class in response to user input;

wherein during execution of the graphical program the second hardware device node is operable to invoke the method.

25. (Original) The memory medium of claim 23, further comprising program instructions executable to:

associate the second hardware device node with a property of the hardware device class in response to user input;

wherein during execution of the graphical program the second hardware device node is operable to perform one or more of: 1) getting the property; and 2) setting the property.

26. (Currently Amended) A memory medium comprising program instructions executable to:

display a first hardware device node in a graphical program in response to user input, wherein the graphical program comprises a plurality of interconnected nodes or icons, wherein the plurality of interconnected nodes or icons visually indicate functionality of the graphical program;

associate the first hardware device node with a first hardware device class in response to user input;

select a method or property of the first hardware device class for the first hardware device node in response to user input;

change the first hardware device node to have an association with a second hardware device class in response to user input; and

perform type checking to determine whether the method or property is valid for the second hardware device class, in response to said changing the first hardware device node to have an association with the second hardware device class;

wherein the graphical program is executable by a computer system.

27. (Original) The memory medium of claim 26, further comprising program instructions executable to:

indicate an invalid condition if the method or property is not valid for the second hardware device class.

28. (Original) The memory medium of claim 26, further comprising program instructions executable to:

prevent execution of the graphical program if the method or property is not valid for the second hardware device class.

29. (Original) The memory medium of claim 26,

wherein the first hardware device node has an input terminal for receiving information specifying a hardware device class with which to associate the first hardware device node;

wherein said associating the first hardware device node with the first hardware device class comprises connecting a first wire to the input terminal in response to user input;

wherein said changing the first hardware device node to have an association with a second hardware device class comprises connecting a second wire to the input terminal in response to user input.

30. (Original) The memory medium of claim 26,  
wherein the first hardware device node is a register access node.

31. (Original) The memory medium of claim 26,  
wherein said performing type checking to determine whether the method or property is valid for the second hardware device class comprises:

determining a list of valid methods and properties for the second hardware device class; and

determining whether the method or property is included in the list of valid method and properties.

32. (Currently Amended) A system for propagating type information for hardware device nodes in a graphical program, the system comprising:

a computer including a processor coupled to a memory;

a display screen coupled to the computer;

a user input device coupled to the computer;

wherein the processor is operable to execute program instructions stored in the memory to:

display on the display screen a first hardware device node in the graphical program in response to user input received from the user input device, wherein



the graphical program comprises a plurality of interconnected nodes or icons, wherein the plurality of interconnected nodes or icons visually indicate functionality of the program;

associate the first hardware device node with a hardware device;

display on the display screen a second hardware device node in the graphical program in response to user input received from the user input device;

connect the first hardware device node to the second hardware device node in response to user input received from the user input device;

propagate information from the first hardware device node to the second hardware device node, wherein the information specifies the hardware device with which the first hardware device node is associated, wherein said propagating occurs in response to said connecting the first hardware device node to the second hardware device node;

wherein the graphical program is executable by the computer.

33. (Original) The system of claim 32,

wherein said displaying the first and second hardware device nodes in the graphical program comprises including the first and second hardware device nodes in a block diagram of the graphical program, wherein the block diagram visually indicates functionality of the graphical program.

34. (Original) The system of claim 32, wherein the processor is further operable to execute program instructions stored in the memory to:

associate the second hardware device node with the hardware device with which the first hardware device node is associated, in response to said propagating the information to the second hardware device node.

35. (Original) The system of claim 32,

wherein said associating the first hardware device node with a hardware device comprises associating the first hardware device node with a hardware device class corresponding to the hardware device;

wherein said propagating information from the first hardware device node to the second hardware device node comprises propagating information specifying the hardware device class with which the first hardware device node is associated.

36. (Original) The system of claim 35,

wherein the processor is further operable to execute program instructions stored in the memory to associate the second hardware device node with the hardware device class, in response to said propagating the information to the second hardware device node.

37. (Original) The system of claim 36, wherein the processor is further operable to execute program instructions stored in the memory to:

associate the second hardware device node with a method of the hardware device class in response to user input;

wherein during execution of the graphical program the second hardware device node is operable to invoke the method.

38. (Original) The system of claim 36, wherein the processor is further operable to execute program instructions stored in the memory to:

associate the second hardware device node with a property of the hardware device class in response to user input;

wherein during execution of the graphical program the second hardware device node is operable to perform one or more of: 1) getting the property; and 2) setting the property.

39. (Currently Amended) A system for performing type checking for a hardware device node in a graphical program, the system comprising:

a computer including a processor coupled to a memory;

a display screen coupled to the computer;

wherein the processor is operable to execute program instructions stored in the memory to:

display on the display screen a first hardware device node in [[a]] the graphical program in response to user input, wherein the graphical program comprises a plurality of interconnected nodes or icons, wherein the plurality of interconnected nodes or icons visually indicate functionality of the graphical program;

associate the first hardware device node with a first hardware device class in response to user input;

select a method or property of the first hardware device class for the first hardware device node in response to user input;

change the first hardware device node to have an association with a second hardware device class in response to user input; and

perform type checking to determine whether the method or property is valid for the second hardware device class, in response to said changing the first hardware device node to have an association with the second hardware device class;

wherein the graphical program is executable by the computer.

40. (Original) The system of claim 39,

wherein said performing type checking to determine whether the method or property is valid for the second hardware device class comprises:

determining a list of valid methods and properties for the second hardware device class; and

determining whether the method or property is included in the list of valid method and properties.

---

41. (New) The system of claim 39, wherein the graphical program is interpretable or compilable to generate instructions executable by the computer.

42. (New) The system of claim 39, wherein the graphical program comprises a dataflow diagram.

43. (New) The system of claim 32, wherein the graphical program is interpretable or compilable to generate instructions executable by the computer.

44. (New) The system of claim 32, wherein the graphical program comprises a dataflow diagram.

45. (New) The memory medium of claim 26, wherein the graphical program is interpretable or compilable to generate instructions executable by the computer system.

46. (New) The memory medium of claim 26, wherein the graphical program comprises a dataflow diagram.

47. (New) The memory medium of claim 18, wherein the graphical program is interpretable or compilable to generate instructions executable by the computer system.

03 48. (New) The memory medium of claim 18, wherein the graphical program comprises a dataflow diagram.

49. (New) The method of claim 10, wherein the graphical program is interpretable or compilable to generate instructions executable by the computer.

50. (New) The method of claim 10, wherein the graphical program comprises a dataflow diagram.

51. (New) The method of claim 1, wherein the graphical program is interpretable or compilable to generate instructions executable by the computer.

52. (New) The method of claim 1, wherein the graphical program comprises a dataflow diagram.